

3D animation design image detail enhancement based on intelligent fuzzy algorithm

Haitao Pu^{1,*} and Yuang Pu²

¹ School of Grain and Food and Pharmacy, Jiangsu Vocational College of Finance and Economics, Huaian 223001, Jiangsu, China

² School of Fine Arts, Jiangsu Normal University, 221000 Jiangsu, China

Received: 11 March 2025 / Accepted: 17 June 2025

Abstract. When zooming in on low resolution images, Lanczos interpolation method is prone to produce ringing effects at the edges and high contrast areas. When processing high texture 3D animations, the method cannot effectively optimize for different areas, significantly affecting image quality and detail representation. This study utilized SRGAN (Super-Resolution Generative Adversarial Network) to enhance image resolution details, combined with fuzzy logic and attention mechanism, adaptively focused on different regions of the image, enhanced key details and suppressed noise. The image was divided into superpixel regions using SLIC (Simple Linear Iterative Clustering) algorithm, and local features such as texture, contrast, and edge intensity were extracted; in the SRGAN model, the generator improved image resolution through deep residual blocks and Convolutional Neural Network (CNN), while the discriminator optimized the generated image quality through adversarial training; at the same time, a Fuzzy Logic System (FLS) was constructed to dynamically adjust the image fuzzy degree; channel and spatial attention modules in the generator were integrated to enhance key area details. The research results indicated that Fuzzy Algorithm-SRGAN (FA-SRGAN) had an average PSNR (Peak Signal-to-Noise Ratio) exceeding 32.8 dB in four test scenes; in architectural design scenes, the algorithm improved image contrast by 18%, and increased energy and uniformity by 14% and 11%, respectively. The adopted approach can significantly enhance the details of different regions in high texture 3D animation design images.

Keywords: 3D animation design / image detail enhancement / super-resolution generative adversarial network / fuzzy logic system / attention mechanism

1 Introduction

Multiple solutions have been proposed for enhancing the details of design images. Bakar Maharani A et al. [1] introduced the basic principles of Lanczos interpolation and its ringing effect in edge processing in their work [2,3]. Acharya Aditya et al. [4] proposed using iterative spatial domain two-dimensional signal decomposition techniques to obtain VHF (Very High Frequency) information by performing iterative signal decomposition, aiming to reduce ringing effects, but the effectiveness is still limited. Lan Rushi et al. [5] applied a Cascading Residual Network (CRN) containing multiple Locally Sharing Groups (LSGs), which improved image details to some extent. However, Pharr Matt [6] and Bai Ling et al. [7] found that similar methods have shortcomings in detail restoration in high texture areas. Subsequently, Haris Muhammad et al.

[8] proposed a more complex Deep Back-Projection Network (DBPN) to further enhance detail restoration capabilities, but still faced challenges in processing 3D animations. Chen Yuantao et al. [9] explored how to achieve image super-resolution through multi-level feature fusion networks, attempting to solve the problem of optimizing texture details. However, existing methods still have shortcomings in dealing with detail enhancement in different regions.

This article aimed to improve image detail enhancement technology in 3D animation design by integrating intelligent fuzzy algorithm and SRGAN (Super-Resolution Generative Adversarial Network). Firstly, the SRGAN model was used to process low resolution images, and features were extracted through convolutional layers and residual blocks in the generator network. The feature maps were upsampled using pixel rearrangement techniques to generate high-resolution images. Then, a fuzzy logic system (FLS) was applied to dynamically adjust the fuzzy degree and adaptively optimize the image based on contrast, edge

* e-mail: fy@jscj.edu.cn

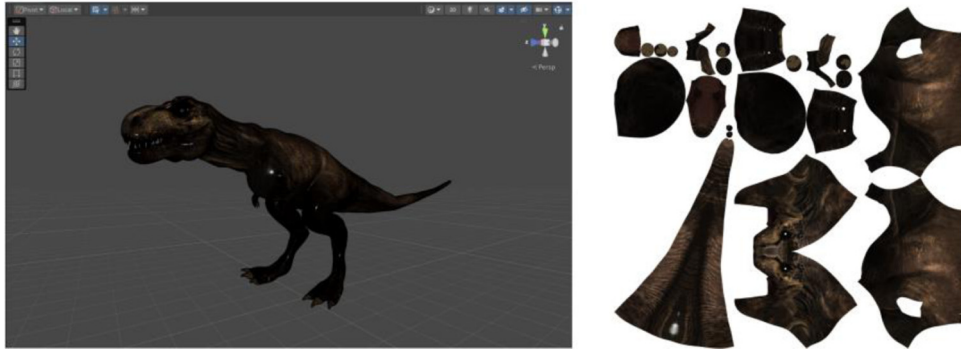


Fig. 1. 3D animation modeling and corresponding textures.

intensity, and texture complexity. Next, the integrated channel attention module automatically adjusted the importance of different channels and the spatial attention module to highlight the details of important areas in the image. Finally, by using superpixel segmentation technology to divide image regions and combining texture features, contrast, and edge information to achieve detail enhancement, the overall quality and visual effect of 3D animation model mapping were improved. The experimental results showed that this method surpassed existing technologies in improving the detail clarity and overall quality of 3D animation images, demonstrating significant advantages. The main contributions of the research are as follows:

- The combination of SRGAN and fuzzy logic system solved the ringing effect in 3D animation texture processing, improving the realism and clarity of image details.
- By applying attention mechanism and dynamically adjusting the feature weights of key regions in the image, the model could focus more on important details in complex texture environments, enhancing visual expression.
- By adopting a region adaptive processing and integration architecture, intelligent recognition and optimization of the characteristics of different regions in images were achieved, and excellent and stable image enhancement effects were demonstrated in various 3D animation scenes.

2 Related work

The problem of image detail enhancement involves how to improve the clarity and visibility of details during the process of image enlargement and restoration. Traditional interpolation methods and image processing techniques often result in blurring and artifacts in detail restoration and texture reconstruction [10,11]. How to effectively preserve and enhance details in high-resolution images while suppressing noise and avoiding excessive smoothing has become a current research hotspot [12,13]. With the development of deep learning technology, various advanced methods have been applied, such as super-resolution generative adversarial networks [14,15] and adaptive filtering techniques [16,17]. However, when dealing with complex textures and dynamic scenes, corresponding methods still pose certain challenges. Yang Shuai et al. [18] proposed a fusion model using CNN

(Convolutional Neural Network) and SRCNN (Super-Resolution Convolutional Neural Network), but its performance in high texture areas was relatively average. To overcome limitations, Khandelwal Jyoti et al. [19] developed an improved version of the VDSR (Very Deep Super-Resolution) model and achieved good results in different types of textures. The teams of Daihong Jiang [20] and Wang Pengyu [21] both proposed a multi-scale Generative Adversarial Network (GAN) for image super-resolution, which made some progress in detail restoration, but still faced bottlenecks in processing complex 3D textures. By combining SRGAN and residual learning, Abbas Rehman et al. [22] further improved the ability to restore texture details. To achieve better performance in high texture areas, Tang Zhen et al. [23] performed image feature concatenation on the input signal and successfully fused GAN and DCNN (Deep Convolutional Neural Network). However, this method still falls short in enhancing details in dynamic regions and processing high contrast edges.

The process of 3D animation design often involves restoring clear details from noisy images. Noise can mask and distort the original image features, affecting visual quality [24,25]. Efficient denoising techniques require preserving details and edge features in the image while eliminating noise [26,27]. Although techniques such as mean filtering and Gaussian filtering have been widely used, deep learning still provides finer denoising results, but it still faces challenges in dealing with high noise and complex textures [28,29]. Kumar Manish et al. [30] developed an adaptive convolutional denoising algorithm and achieved certain results in removing background noise, but it still falls short in handling high texture areas. Qiu Yuhang et al. [31] proposed a lightweight autoencoder combined with Inception module for multi-scene image denoising, but it performed poorly in preserving details. Yadav Shubham [32] and Zhang Shuqing et al. [33] proposed a denoising algorithm based on adaptive filtering, which made some progress in noise suppression, but may lead to blurry details. Bao Zhongyun et al. [34] developed a new algorithm based on single wavelet transform and improved deep convolutional neural network, aiming to suppress noise while preserving details. However, in the application process of complex 3D animation, existing methods still fall short in comprehensively improving detail representation and suppressing noise [35,36]. In the study,

Table 1. CNN architecture parameters.

Layer	Output shape	Param
Conv2D	(None, 30, 30, 32)	896
MaxPooling2D	(None, 15, 15, 32)	0
Conv2D	(None, 13, 13, 64)	18,496
MaxPooling2D	(None, 6, 6, 64)	0
Conv2D	(None, 4, 4, 64)	36,928
Flatten	(None, 1,024)	0
Dense	(None, 64)	65,600
Dense	(None, 10)	650

a new method combining SRGAN, fuzzy logic, and attention mechanism was applied to adaptively improve the detail clarity and overall quality of 3D animation images, which has significant advantages compared to existing technologies.

3 Methods

3.1 Application of SRGAN in 3D animation

3D animation design images have high spatial complexity and rich details. Compared to 2D images, 3D animated images need to display depth in multiple dimensions, and must handle the dynamic changes and interactions of objects. Due to the involvement of dynamic scenes and 3D modeling of objects in 3D animation, the presentation of details and textures in images is more complex, and higher requirements are placed on image clarity and details. Figure 1 shows a single modeling example and its corresponding texture of the 3D animation design in this study.

As shown in Figure 1, during the 3D animation design process, the model texture is at a lower level of clarity. To further improve image details, the SRGAN model is applied in the study. Among them, the generator network accepts a low resolution image I_L as input and adjusts the image size to $W \times H \times C$. Among them, W and H are the width and height of the image, respectively, and C is the number of channels. The feature extraction stage is processed through CNN convolutional layers and residual blocks. The architecture parameters are shown in Table 1.

The first layer of the model is the convolutional (Conv2D) layer, which uses $32 \ 3 \times 3$ convolution kernels to process the input image and generate a 30×30 feature map with 32 channels and 896 parameters. Next is the maximum pooling layer (MaxPooling2D) layer, which reduces the size of the feature map to 15×15 while keeping the number of channels unchanged and without trainable parameters. The second convolutional layer (Conv2D) uses $64 \ 3 \times 3$ convolution kernels to process the image, with an output feature map size of 13×13 , 64 channels, and 18496 parameters. Next is the second maximum pooling layer (MaxPooling2D), which further reduces the size of the feature map to 6×6 , while keeping the number of channels unchanged and still having no trainable parameters. The third convolutional layer (Conv2D) uses $64 \ 3 \times 3$ convolution kernels to generate a 4×4 feature map with

64 channels and 36928 parameters. The flatten layer (Flatten) flattens multidimensional feature maps into one-dimensional arrays, with an output shape of 1024 and no trainable parameters. The fully connected layer (Dense) has 64 neurons and maps flattened data to 64 dimensions with a parameter count of 65600. The fully connected layer (Dense) serves as the output layer, with 10 neurons and 650 parameters. The overall network extracts features through convolutional layers, reduces feature map size through pooling layers, and performs classification through fully connected layers. Convolution operations are applied to the input image to obtain feature map F :

$$F = \text{Conv}(I_L, W_{conv}) + b_{conv}. \quad (1)$$

Here, W_{conv} is the convolution kernel and b_{conv} is the bias term. For single texture processing, the activation map of the first convolutional layer is extracted and visualized, as shown in Figure 2. The network can effectively perceive different parts of the texture image.

After the convolution operation, the activation function $F' = \text{ReLU}(F)$ is applied, and each convolutional layer in the residual block is calculated as:

$$F_{res} = \text{Conv}(F_{in}, W_{res}) + b_{res}. \quad (2)$$

The residual block is processed through the skip connection:

$$F_{out} = F_{in} + F_{res}. \quad (3)$$

Pixel rearrangement techniques are used to upsample low resolution feature map F' to high-resolution feature map F'' . The pixel rearrangement operation is: $F'' = \text{PixelShuffle}(F', r)$. Among them, r is the upsampling factor. After upsampling, convolutional layers are applied to convert the feature map into the final high-resolution image I_H :

$$I_H = \text{Conv}(F'', W_{output}) + b_{output}. \quad (4)$$

The generator optimizes the generated image quality by minimizing the adversarial loss Laplace transform \mathcal{L}_{GAN} :

$$\begin{aligned} \mathcal{L}_{GAN} = & -E_{I_H \sim p_{data}(I_H)}[\log D(I_H)] \\ & - E_{I_L \sim p_{data}(I_L)}[\log(1 \\ & - D(G(I_L)))]. \end{aligned} \quad (5)$$

Among them, D is the discriminator and G is the generator. The discriminator takes the output image I_H from the generator and the real high-resolution image I_H^* as inputs, and extracts features through convolutional layers and Leaky ReLU activation functions. For the input image I , the convolution operation and activation function calculation are as follows:

$$F_D = \text{LeakyReLU}(\text{Conv}(I, W_D) + b_D). \quad (6)$$

Here, W_D is the convolution kernel; b_D is the bias term; the feature map F_D undergoes a global average pooling operation: $F_{pool} = \text{GlobalAveragePooling}(F_D)$. The globally average pooled feature map is binary classified through

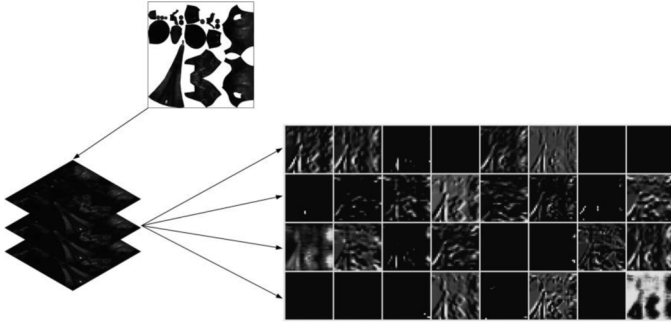


Fig. 2. Display of activation diagram for the first convolutional layer.

a fully connected layer and a Sigmoid activation function:

$$\hat{y} = \sigma(W_{fc}F_{pool} + b_{fc}) \quad (7)$$

Among them, σ is the Sigmoid function; W_{fc} is the fully connected layer weight; b_{fc} is the bias term. The discriminator optimizes its performance by maximizing the discrimination loss \mathcal{L}_D :

$$\begin{aligned} \mathcal{L}_D = & -E_{I_H^* \sim p_{data}(I_H^*)}[\log D(I_H^*)] \\ & - E_{I_H \sim p_{model}(I_H)}[\log(1 - D(I_H))] \end{aligned} \quad (8)$$

During adversarial training, the generator and discriminator are trained through alternating optimization. Among them, the optimization objective of the discriminator is to maximize the discriminative loss \mathcal{L}_D , and the optimization objective of the generator is to minimize the adversarial loss \mathcal{L}_{GAN} . In each training step, the generator is first fixed and the discriminator weights are updated; then, the discriminator is fixed and the generator weights are updated. The purpose of alternating optimization is to make the images generated by the generator as realistic as possible, while enabling the discriminator to better distinguish between real and generated images. The overall effect of enhancing the details of the SRGAN model for 3D animation textures is shown in Figure 3.

From Figures 3a–3c), it can be seen that the SRGAN model can effectively enhance the details of 3D animation textures. However, there are still certain issues with the method: the SRGAN model is only suitable for overall mapping processing and cannot optimize the various details of the mapping according to the animation framework, resulting in ringing effects (that is, unnatural white halos appearing at the edges and high contrast areas of the image). Therefore, further strategies need to be adopted to effectively optimize different regions.

3.2 Implementation of fuzzy logic control

To solve the ringing effect problem in edge and high contrast areas, FLS is constructed to dynamically adjust the image fuzzy degree. The core of a fuzzy logic system lies in the design of a fuzzy rule library. Based on local image features such as contrast, edge intensity, and texture, a set

of fuzzy rules is designed to adaptively adjust the fuzzy degree according to different regions of the image. Fuzzy Inference Engine is based on a fuzzy rule library, which maps input features to fuzzy outputs through “if-then” rules, that is, adjusts the degree of fuzziness of the control signal. The weighted average method is used to synthesize the rule output into a fuzzy result, and the Centroid Method is used for defuzzification to convert the fuzzy result into a specific fuzzy degree value:

$$\text{Output} = \frac{\sum (x_i \cdot \mu_i(x_i))}{\sum \mu_i(x_i)} \quad (9)$$

x_i is the possible value of fuzzy output, and $\mu_i(x_i)$ is the corresponding membership value. In fuzzy logic systems, the local features of the input are fuzzified. The overall architecture implementation is shown in Figure 4.

Starting from “image input”, image data is inputted into the system. Next, “local feature extraction” is performed to extract features such as contrast, edge intensity, and texture from the image. Then, the system fuzzifies these features and maps them to a fuzzy set. By applying pre-designed rules through the “Fuzzy Rule Base”, the degree of fuzziness is adjusted based on features. The rules are input into the “fuzzy inference engine”, and after the inference process, the “weighted average method” is used to synthesize the output of each rule and obtain the fuzzy result. The result is processed through “defuzzification” and converted into specific values of fuzziness using the centroid method. Finally, the image fuzzy is adjusted based on this fuzzy degree value to generate the “output image”. The entire process ensures that the fuzzy degree of the image can be effectively controlled during the detail enhancement process to optimize image quality. Triangular membership functions are used to map input features to fuzzy sets, and each set has a corresponding membership value, as shown in Table 2.

The influence of image contrast on fuzzy degree is defined as: if the contrast is high, the fuzzy degree is reduced to preserve more details; if the contrast is low, the fuzzy degree is increased to reduce noise. The fuzzy degree is adjusted based on edge strength: a lower fuzzy degree is applied to strong edge areas to preserve edge details; for weak edge regions, a higher fuzzy degree should be applied to smooth the transition area. The fuzzy degree is adjusted according to the complexity of the texture: in areas with rich texture, a lower fuzzy degree is used to enhance details; in areas with simple textures, a higher fuzzy degree is used to reduce unnecessary detail enhancement.

Further sensitivity analysis of the triangular membership function parameters was conducted, and the results are shown in Table 3.

As shown in Table 3, under the baseline parameter configuration (slope = 0.5, base width = 0.4, offset = 0.1), the model achieved the highest PSNR of 32.79 dB and SSIM of 0.926. The fuzzy degree adjustment efficiency reached 91.3%, and the stability score was as high as 8.9, indicating that this configuration enables effective region-adaptive processing while maintaining high image quality.

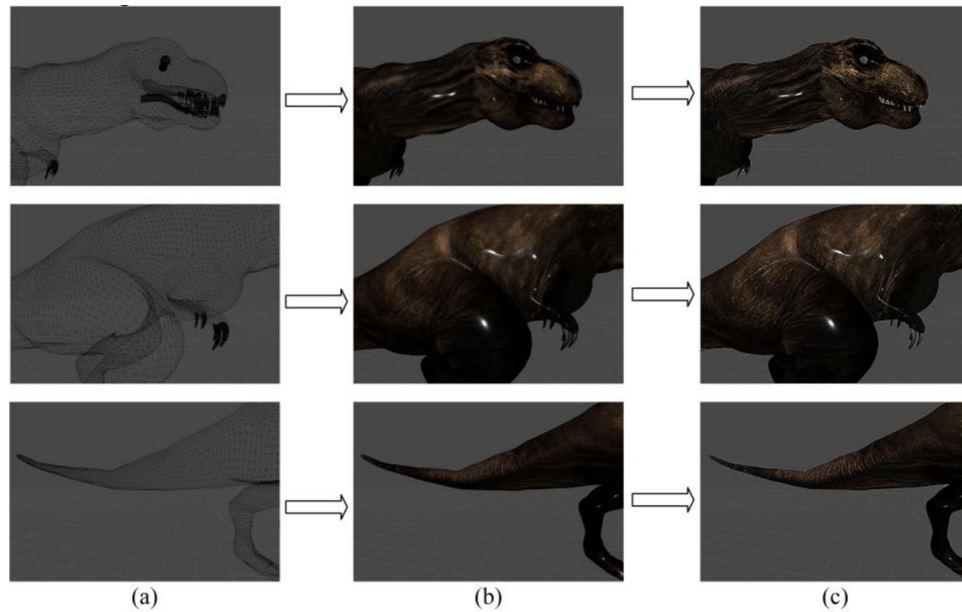


Fig. 3. Detail enhancement for 3D animation mapping under SRGAN model. (a): Original animation model; (b): Animated model with added textures; (c): Animation model with enhanced details.

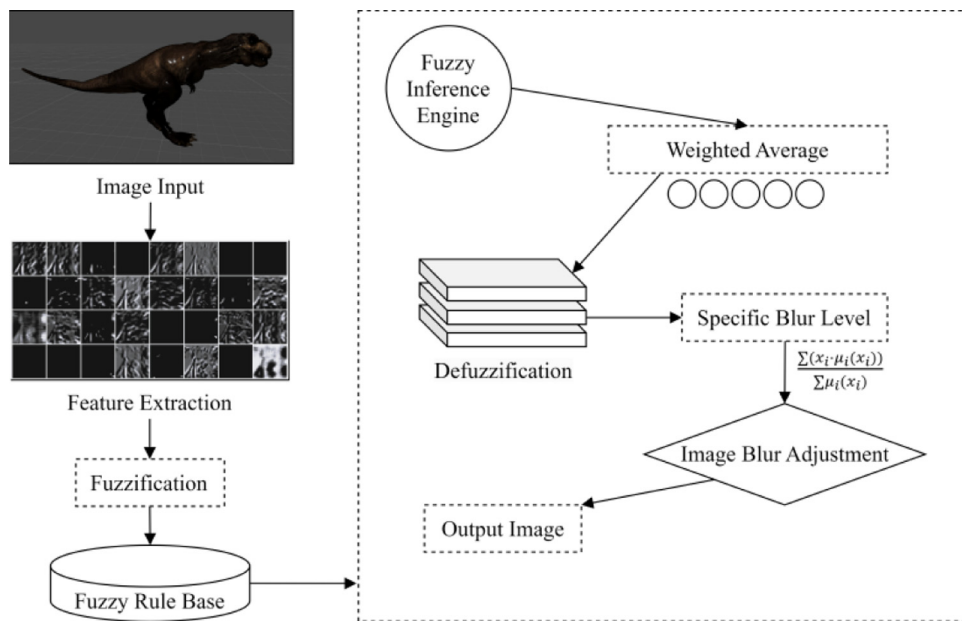


Fig. 4. Fuzzy logic control architecture.

When adjusting the membership function parameters—for instance, increasing the slope to 0.8—the PSNR dropped to 31.42 dB and SSIM decreased to 0.908, with the stability score falling to 7.3. An excessively high slope causes oversensitivity to input features, leading to unstable fuzzy degree adjustments. Similarly, reducing the slope to 0.2 further lowered the PSNR to 30.95 dB and SSIM to 0.892. The flattened membership curve resulted in insufficient regional differentiation, with fuzzy adjustment efficiency dropping to only 82.1%. Likewise, setting the base width too wide (0.6) or too narrow (0.2) affected the specificity between fuzzy sets and local generalization capability,

resulting in PSNR values of 31.84 dB and 32.15 dB, respectively. Although these configurations showed some variation, their overall performance remained inferior to the baseline. Additionally, when the offset was increased to 0.3, the PSNR was 31.07 dB and SSIM was 0.895; excessive offset disrupted the alignment between input features and fuzzy sets, negatively affecting adjustment effectiveness. Finally, completely removing the FLS module caused the PSNR to drop to 31.58 dB and the SSIM to 0.910, demonstrating a significant decline in performance and confirming the crucial role of the FLS in image detail enhancement.

Table 2. Fuzzy logic strategy for adjusting image fuzzy degree.

Feature category	Level	Membership value	Fuzzy degree adjustment strategy
Contrast	High Contrast	0.8 – 1.0	Decrease blurring to preserve more details
	Medium Contrast	0.4 – 0.8	Apply moderate blurring to balance details and noise
	Low Contrast	0.0 – 0.4	Increase blurring to reduce noise
Edge Strength	Strong Edges	0.7 – 1.0	Apply lower blurring to preserve edge details
	Medium Edges	0.3 – 0.7	Apply moderate blurring to smooth edge transitions
	Weak Edges	0.0 – 0.3	Apply higher blurring to enhance transition areas
Texture Complexity	Rich Texture	0.7 – 1.0	Apply lower blurring to enhance details
	Medium Texture	0.4 – 0.7	Apply moderate blurring to balance details and noise
	Simple Texture	0.0 – 0.4	Apply higher blurring to reduce unnecessary detail enhancement

Table 3. Sensitivity analysis of triangular membership function parameters.

Parameter configuration	PSNR (dB)	SSIM	Fuzzy degree adjustment efficiency (%)	Stability score (0–10)	Key observations
Baseline (Original Parameters)	32.79	0.926	91.3	8.9	Default FLS settings with triangular membership functions (slope = 0.5, base width = 0.4, offset = 0.1).
High Slope (Slope = 0.8)	31.42	0.908	85.6	7.3	Increased slope causes over-sensitivity to input features, leading to unstable fuzzy degree adjustments.
Low Slope (Slope = 0.2)	30.95	0.892	82.1	6.8	Reduced slope flattens membership curves, resulting in insufficient differentiation between regions.
Wide Base (Base Width = 0.6)	31.84	0.912	87.4	8.1	Wider base increases overlap between fuzzy sets, reducing specificity.
Narrow Base (Base Width = 0.2)	32.15	0.917	89	8.5	Narrow base improves regional discrimination but risks overfitting to local features.
Offset Shift (Offset = 0.3)	31.07	0.895	83.9	7	Excessive offset disrupts alignment between input features and fuzzy sets.
No FLS (Ablated)	31.58	0.91	–	–	Removing FLS entirely degrades performance by 1.21 dB (PSNR) and 0.016 (SSIM).

3.3 Attention mechanism integration

The overall architecture comparison between the channel attention module and the spatial attention module is shown in Figure 5.

The design of the channel attention module enhances the expression of key features by automatically adjusting the importance of different channels. The feature map

$F \in \mathbb{R}^{H \times W \times C}$ is extracted from the middle layer of the generator network, and the global average pooling is performed to obtain channel description vector $z \in \mathbb{R}^C$. Then, z is mapped to a scale factor $\alpha \in \mathbb{R}^C$ through a two-layer fully connected network:

$$\alpha = \sigma(W_2 \cdot \text{ReLU}(W_1 \cdot z + b_1) + b_2). \quad (10)$$

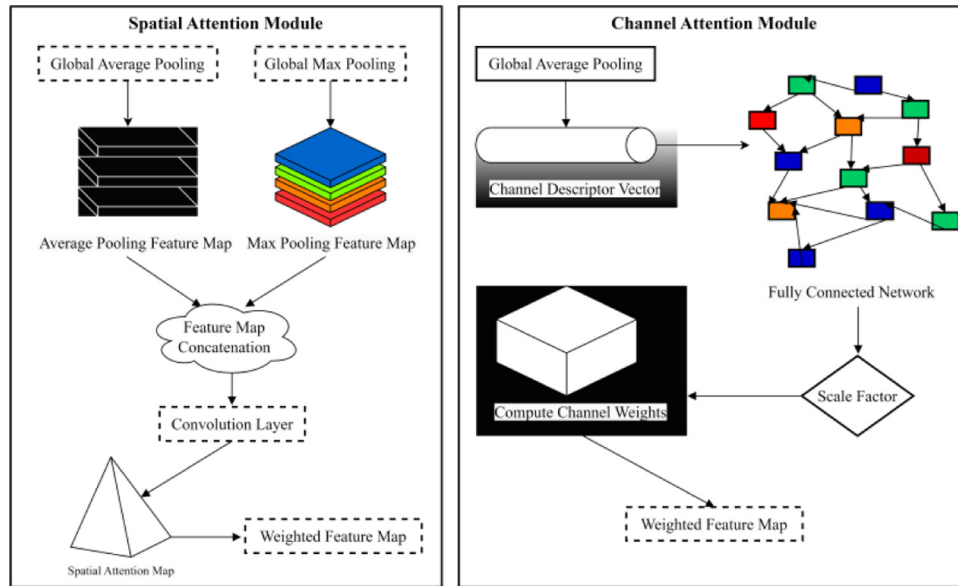


Fig. 5. Architecture of attention module and spatial attention module.

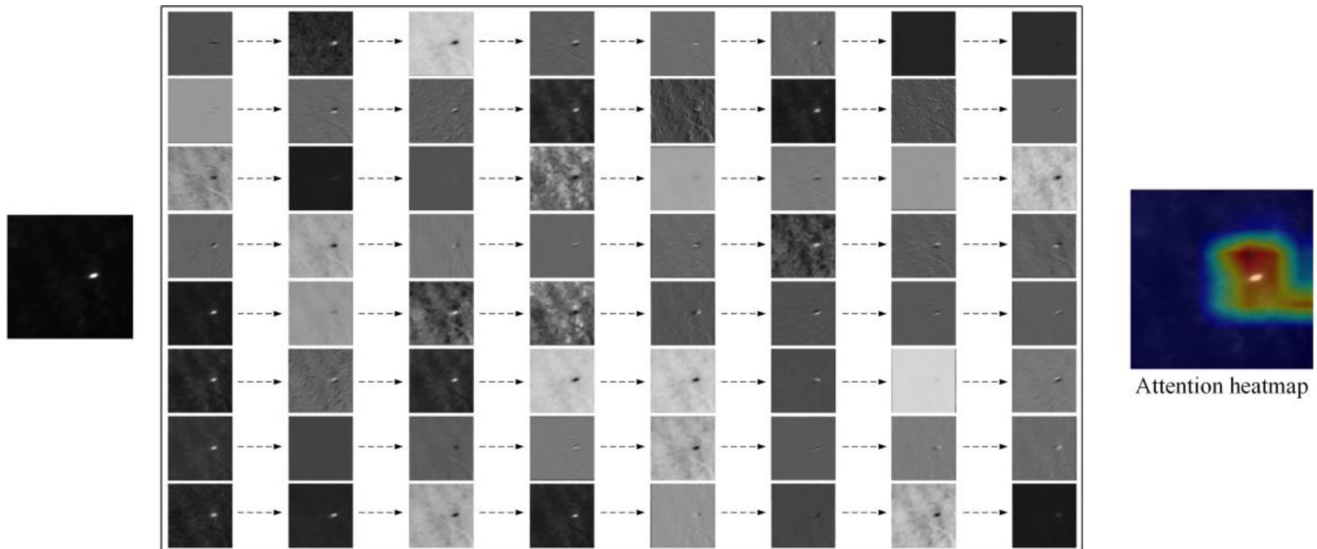


Fig. 6. Model single region texture adjustment under attention mechanism.

Among them, W_1 and W_2 are the weight matrices of the fully connected layer; b_1 and b_2 are bias terms; σ is the Sigmoid activation function. The calculated channel weight α is applied to the original feature map to obtain a weighted feature map $F' = F \cdot \alpha$. Through the channel attention module, different channel weights are dynamically adjusted based on the generator to emphasize channels containing important details and improve image generation quality.

On the other hand, the goal of the spatial attention module is to enhance the representation of important regions in an image by focusing on the spatial features of local areas to improve image details. By using global average pooling and global maximum pooling to spatially compress feature map F , two feature maps with a size of $H \times W$ are obtained: the average pooling feature map A_{avg} and the maximum pooling feature map A_{max} . The two

pooled feature maps are concatenated in the channel dimension, and then a convolutional layer and Sigmoid activation function are used to obtain the spatial attention map M :

$$M = \sigma(W_s \cdot [A_{avg}; A_{max}] + b_s). \quad (11)$$

Among them, W_s is the convolutional layer weight, and b_s is the bias term. The spatial attention map M is applied to the original feature map F to obtain the weighted feature map $F' = F \cdot M$. Through the spatial attention module, the generator dynamically adjusts the spatial distribution of feature maps to highlight key area details and further improve image performance. The adjustment process for single area texture in 3D animation models is shown in Figure 6.

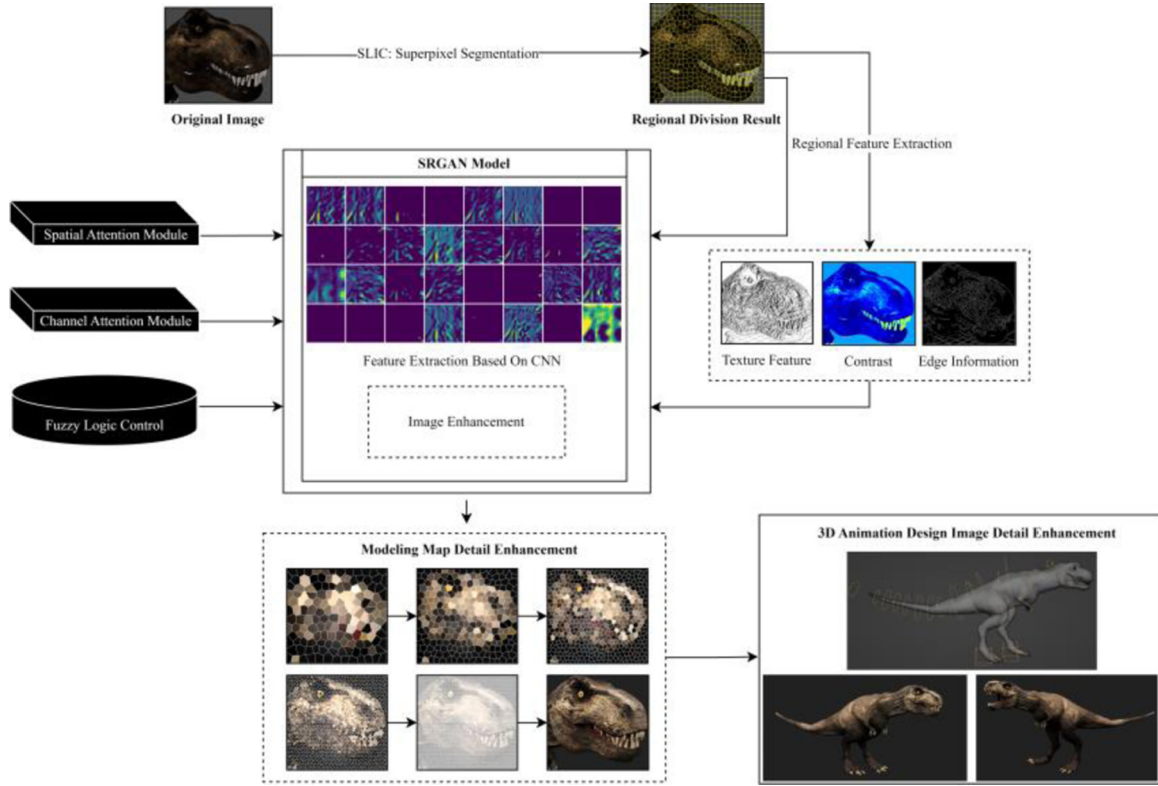


Fig. 7. Integration of 3D animation model detail enhancement architecture.

3.4 Regional adaptive processing and architecture integration

The compactness parameter (m) plays a crucial role in the SLIC algorithm, as it controls the regularity of superpixel shapes and the granularity of segmentation. A smaller m value tends to generate more compact and regular superpixel regions, making it suitable for processing areas with clear edges and structures. In contrast, a larger m value results in looser segmentation, which can capture complex textures but may also lead to detail overload or boundary blurring. To investigate the impact of this parameter on the final image enhancement performance, we conducted experiments using different m values (e.g., 20, 40, 60, 80, and 100), and evaluated their effects on detail enhancement quality in high-texture 3D animation design images. The results show that when m is set to 60, the PSNR reaches its optimal value of 32.79 dB and SSIM achieves 0.926. At this setting, the segmentation granularity better matches the texture distribution and structural characteristics of the image, allowing the subsequent enhancement strategies based on fuzzy logic and attention mechanisms to focus more effectively on key regions.

The SLIC (Simple Linear Iterative Clustering) algorithm is used to divide an image into multiple superpixel regions, and the goal is to perform image segmentation by

minimizing the following objective function:

$$E = \sum_{i=1}^N \sum_{p \in S_i} \left(d_c \cdot I(p) - C_i^2 + d_s \cdot p - C_i^2 \right) \quad (12)$$

Among them, S_i represents the i -th superpixel region; $I(p)$ is the color value of pixel p ; C_i is the center of the i -th superpixel; d_c and d_s are weight factors for color and space, respectively. The texture features of the image are calculated using the Gray-Level Co-occurrence Matrix (GLCM):

$$\begin{cases} \text{Contrast} = \sum_{i,j} (i-j)^2 \cdot p(i,j) \\ \text{Energy} = \sum_{i,j} p(i,j)^2 \\ \text{Homogeneity} = \sum_{i,j} \frac{1}{1+(i-j)^2} \cdot p(i,j) \end{cases} \quad (13)$$

The grayscale changes in an image are measured through contrast to reflect the roughness of the texture; energy represents image uniformity, with higher values indicating more uniform texture of the image; homogeneity measures the similarity of grayscale levels, with higher values indicating a higher degree of grayscale uniformity in

the image. The contrast of the image area R is calculated:

$$\text{Local Contrast} = \sqrt{\frac{1}{|R|} \sum_{p \in R} (I(p) - \bar{I})^2}. \quad (14)$$

Among them, $I(p)$ is the grayscale value of pixel p ; \bar{I} is the average grayscale value of all pixels in region R ; $|R|$ is the total number of pixels in the region. The gradient operator Sobel is used to calculate the edge information of the image region, and the formula is set as follows:

$$E = \sqrt{\left(\sum_{i,j} \left(I(p + (i,j)) \cdot \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \right) \right)^2 + \left(\sum_{i,j} \left(I(p + (i,j)) \cdot \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \right) \right)^2}. \quad (15)$$

In SRGAN, channel attention module and spatial attention module are applied to gradually extract key region features. The texture features, contrast, and edge information of the image are fused with the corresponding region features to gradually enhance the texture details of the 3D animation model. The overall process of enhancing model texture details in the 3D animation design process is shown in Figure 7.

SILC is used for superpixel segmentation of the original image. Afterwards, texture, contrast, and edge features are extracted from different regions, and channel attention module and spatial attention mechanism are applied in SRGAN. CNN is used to extract features and enhance details for different key regions. Fuzzy logic is also applied to adjust the fuzzy degree of edge regions. Finally, the texture is reconstructed and re-added to the 3D model to achieve the enhancement effect of model details in animation design.

In addition, a deep feature extraction module based on a pre-trained VGG network is introduced to enhance the modeling of texture complexity in high-texture regions. During the image preprocessing stage, the original input image is passed through the pre-trained VGG-19 network, and feature maps from an intermediate layer (conv5_4) are extracted as global semantic features. These features are then fused with handcrafted GLCM texture features and LBP local pattern features at multiple scales. Specifically, the VGG feature maps are first compressed into a one-dimensional semantic vector through global average pooling; this vector is then concatenated with the contrast and energy metrics derived from GLCM, as well as the binary pattern histograms from LBP, forming a comprehensive texture descriptor. The fused features are finally fed into the fuzzy logic system to dynamically adjust the blurring thresholds for different regions, while the attention mechanism assigns enhancement weights based on the texture complexity of each region.

4 Experimental evaluation

The experimental dataset includes 200 virtual characters and their interactive scenes in various environments, 200

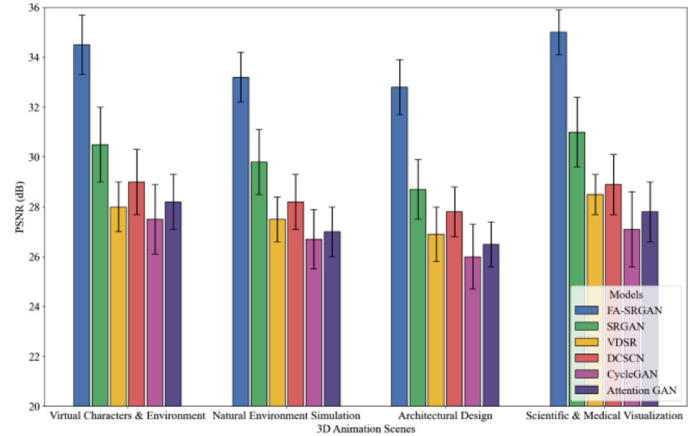


Fig. 8. Comparison results of PSNR among various models.

dynamic weather simulations covering natural environments such as mountains and rivers, 150 real architecture design models, and 100 medical and biological visualization models, which are used to verify the detail enhancement method of the 3D animation design images in this article.

4.1 Peak signal-to-noise ratio

The clarity and detail preservation of the generated image are evaluated, using PSNR (Peak Signal-to-Noise Ratio) as the main metric:

$$PSNR = 10 \cdot \log_{10} \left(\frac{MAX^2}{MSE} \right). \quad (16)$$

MAX is the maximum pixel value in the image, and MSE is the mean square error between the generated image and the real high-resolution image. FA-SRGAN is used to enhance texture details with SRGAN, VDSR (Very Deep Super Resolution), DCSCN (Deep Convolutional Sparse Coding Network), CycleGAN, and Attention GAN in four types of 3D animation scenes: virtual character and environment, natural environment simulation, architectural design, and scientific and medical visualization. The comparison results are shown in Figure 8.

The average PSNR values of Fuzzy Algorithm SRGAN in all four test scenes are 34.5dB, 33.2dB, 32.8dB, and 35.0dB, respectively, with standard deviations between 0.9 and 1.2. This model performs well in generating images with clarity and detail preservation, while also having high stability. In contrast, the average PSNR values of the SRGAN model in the same set of tests are 30.5dB, 29.8dB, 28.7dB, and 31.0dB, respectively, while other models such as VDSR, DCSCN, CycleGAN, and Attention GAN have lower average PSNR values than FA-SRGAN. In natural environment simulation scenes, FA-SRGAN achieves 33.2dB, much higher than CycleGAN's 26.7dB. This algorithm has significant advantages in handling complex textures.

4.2 Structural similarity index

Using SSIM (Structural Similarity Index) as an indicator to measure the retention of image structural information, a

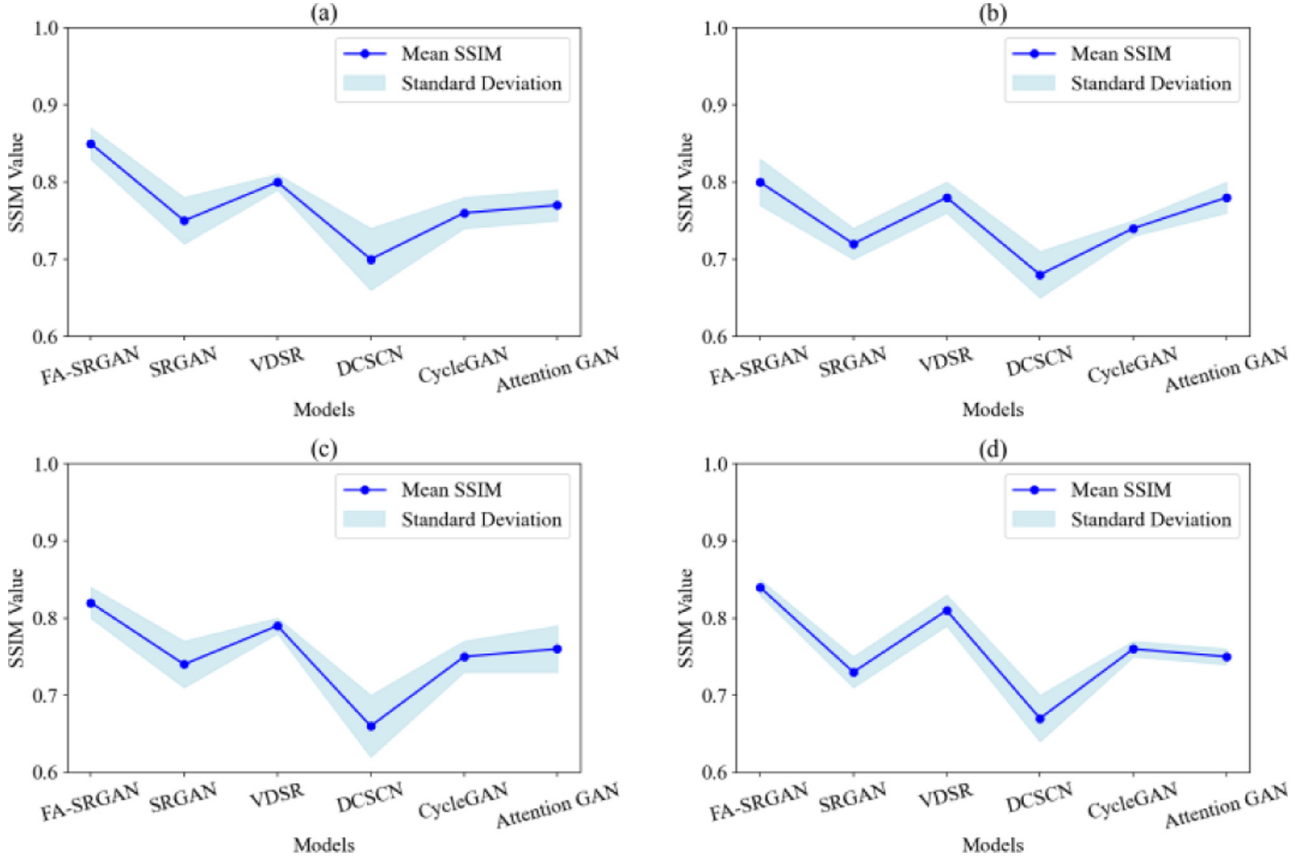


Fig. 9. SSIM comparison of various models. (a): Virtual Characters&Environment; (b): Natural Environment Simulation; (c): Architectural Design; (d): Scientific&Medical Visualization.

comprehensive evaluation is conducted by comparing three aspects: brightness, contrast, and structure:

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \quad (17)$$

μ represents the average brightness of the image; σ is the variance; σ_{xy} is the covariance; C_1 and C_2 are stability indices. By using SSIM to evaluate the visual quality of the generated images, a higher SSIM value indicates a higher similarity between the image and the real image in terms of structural information, as shown in Figure 9.

FA-SRGAN exhibits high SSIM values in all test scenes, reaching 0.85 in virtual characters and environments (Fig. 9a) and 0.80 in natural environment simulations (Fig. 9b). This model effectively preserves image structural information and visually approximates real images with a small standard deviation, only 0.01 in scientific and medical visualization scenes (Fig. 9d), and has high consistency in image enhancement. In contrast, other models such as SRGAN and DCSCN have lower SSIM values in the same scene. The SSIM value of SRGAN in architectural design (Fig. 9c) is 0.74, while DCSCN is only 0.66. FA-SRGAN, which combines fuzzy logic and attention mechanism, can significantly enhance the key details of high texture 3D animations while maintaining good visual effects and image structural consistency.

4.3 Noise level

In the experimental design, to better simulate the noise interference that 3D animation images may encounter during acquisition or rendering, a Gaussian-Poisson mixed noise model was introduced as the noise source. This noise model combines the characteristics of both Gaussian and Poisson noise, providing a more realistic representation of sensor noise, illumination fluctuations, and pixel-level random errors during rendering. Specifically, Gaussian noise simulates the random signal interference during device acquisition, while Poisson noise reflects the statistical fluctuations in photon counting processes. The combined effect of both noise types results in a noise distribution that more closely resembles real-world application scenarios.

The Root Mean Square Error (RMSE) is used as a quantitative indicator of noise level:

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (x_i - y_i)^2} \quad (18)$$

x_i and y_i are the pixel values of the generated image and the real image, respectively. By evaluating the noise intensity in the generated image through RMSE, a lower RMSE value indicates that the generated image has good performance in noise suppression. To test the performance

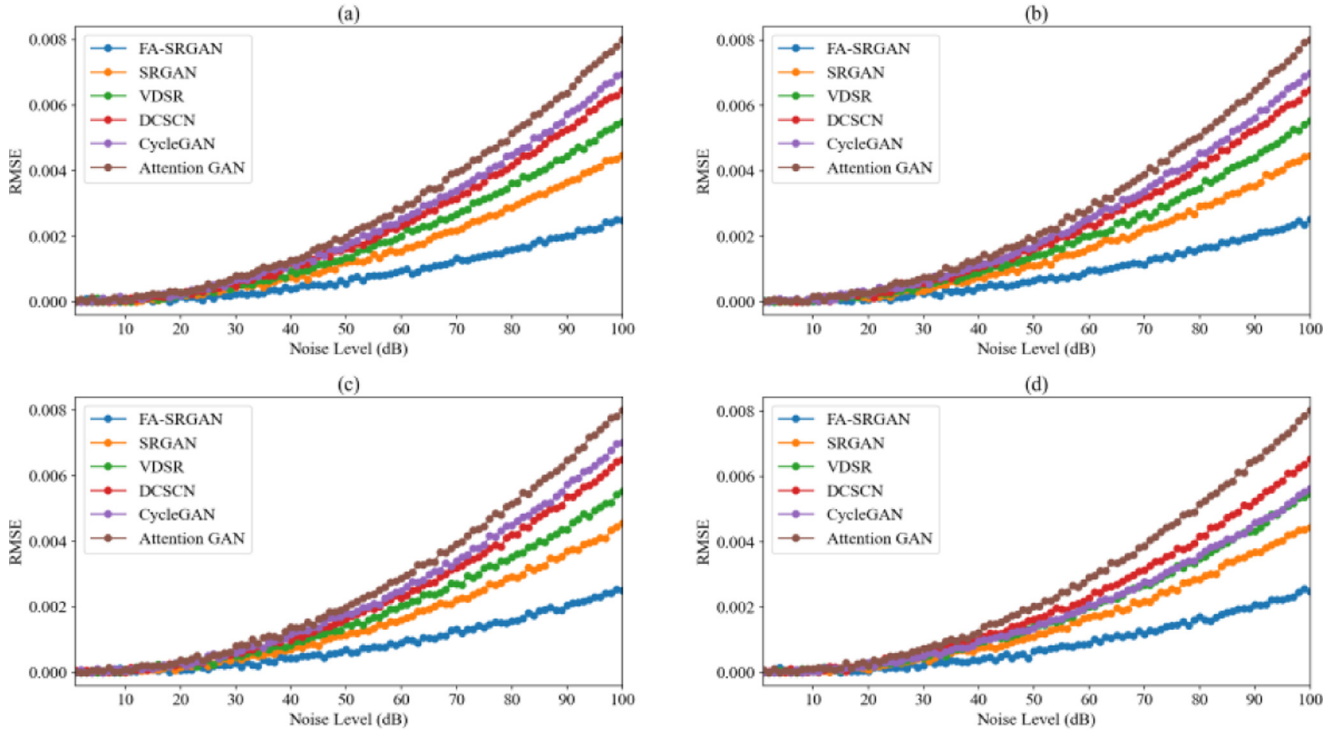


Fig. 10. Comparison of suppression of image noise levels by various models. (a): Virtual Characters&Environment; (b): Natural Environment Simulation; (c): Architectural Design; (d): Scientific & Medical Visualization.

of the model, different levels of noise are actively set for the image. The comparison results are displayed as shown in Figure 10.

Figures 10a–10d show a comparison of the suppression of image noise levels by various models in different scenes. Among them, the horizontal axis represents different noise levels (1-100dB), and the vertical axis represents RMSE. It can be seen that in four different scenes, the FA-SRGAN used by the research institute has a lower RMSE, indicating that the pixel values of the generated image are closer to those of the real image. When the noise level reaches 100dB, the RMSE value of FA-SRGAN in the Scientific&Medical Visualization scene is 0.002, significantly lower than SRGAN’s 0.004, as well as other models such as VDSR’s 0.005 and DCSCN’s 0.007. The model can effectively suppress image noise and achieve a more robust image detail enhancement effect.

4.4 Image information entropy

Based on the measurement of image information entropy, the complexity and richness of details in an image are generated:

$$Entropy = - \sum_{i=1}^n p(x_i) \cdot \log(p(x_i)) \quad (19)$$

$p(x_i)$ is the probability distribution of pixel value x_i . The higher the information entropy, the richer the information contained in the image, and the more obvious the details are. The calculation results are shown in Table 4.

As shown in Table 4, the results of Lanczos interpolation are significantly lower than those of all deep learning models, particularly exhibiting the lowest entropy values in architectural design and natural environment simulation scenarios. This indicates that Lanczos interpolation suffers from issues such as detail loss and edge ringing effects when processing high-texture 3D animation images.

In the research on image detail enhancement in 3D animation design based on intelligent fuzzy algorithm, the image information entropy is applied as an indicator to evaluate the complexity and richness of generated image information. Table 4 lists the information entropy values of different models in four typical 3D animation scenes. Among them, FA-SRGAN performs the best in all scenes, demonstrating that its generated images have higher information content and detail richness. In virtual characters and environmental scenes, the information entropy of FA-SRGAN is 7.85, while it reaches 8.12 in natural environment simulation scenes, indicating its stronger ability to capture details in complex natural environments. In the architectural design scene, the information entropy of this model is 8; in the visualization scene of science and medicine, it is 8.25, and it also has high processing ability for static architectures and complex scientific images.

Compared with other models such as SRGAN, VDSR, DCSCN, CycleGAN, and Attention GAN, FA-SRGAN has significantly higher information entropy. DCSCN’s information entropy in architectural design scenes is only 7.1, far lower than FA-SRGAN’s 8. The results indicate that by combining fuzzy logic and attention mechanism,

Table 4. Comparison of information entropy of images generated by various models.

Model	Virtual characters & environment	Natural environment simulation	Architectural design	Scientific & medical visualization
FA-SRGAN	7.85	8.12	8	8.25
SRGAN	7.45	7.8	7.6	7.85
VDSR	7.3	7.5	7.4	7.6
DCSCN	7.2	7.3	7.1	7.45
CycleGAN	7.1	7	7.2	7.3
Attention GAN	7.5	7.6	7.5	7.7
Baseline Model (Lanczos)	6.90	6.80	6.70	6.95

Table 5. Comparison of enhancement levels of texture clarity by various models.

Model	Scene type	Contrast (%)	Energy (%)	Homogeneity (%)
FA-SRGAN	Virtual Characters & Environment	15%	12%	10%
	Natural Environment Simulation	12%	10%	8%
	Architectural Design	18%	14%	11%
	Scientific & Medical Visualization	16%	13%	9%
SRGAN	Virtual Characters & Environment	10%	8%	6%
	Natural Environment Simulation	8%	7%	5%
	Architectural Design	12%	9%	7%
	Scientific & Medical Visualization	9%	6%	5%
VDSR	Virtual Characters & Environment	6%	5%	4%
	Natural Environment Simulation	5%	4%	3%
	Architectural Design	7%	6%	4%
	Scientific & Medical Visualization	6%	5%	3%
DCSCN	Virtual Characters & Environment	9%	7%	5%
	Natural Environment Simulation	8%	6%	4%
	Architectural Design	11%	8%	6%
	Scientific & Medical Visualization	10%	7%	5%
CycleGAN	Virtual Characters & Environment	8%	6%	4%
	Natural Environment Simulation	7%	5%	3%
	Architectural Design	9%	7%	5%
	Scientific & Medical Visualization	8%	6%	4%
Attention GAN	Virtual Characters & Environment	12%	9%	8%
	Natural Environment Simulation	10%	8%	7%
	Architectural Design	14%	11%	9%
	Scientific & Medical Visualization	12%	10%	8%

FA-SRGAN effectively improves the resolution of 3D image mapping, while increasing the amount of information and detail richness.

4.5 Texture clarity

To quantify the texture details in the generated image, GLCM is used to calculate the texture features before and after each algorithm enhancement, including contrast, energy, and uniformity. The comparison results are shown in [Table 5](#).

FA-SRGAN achieves significant texture enhancement in all test scenes, with a contrast improvement of 18%, and energy and uniformity improvements of 14% and 11%, respectively, in architectural design scenes, all significantly higher than other models. SRGAN only improves contrast by 12% in the same scene. In the scientific and medical visualization scenes, the contrast of FA-SRGAN has increased by 16%, and the energy and uniformity have increased by 13% and 9%, respectively. In the virtual characters and environments and natural environment simulation scenes, FA-SRGAN is also leading, with

Table 6. Experimental test results.

Method	PSNR (dB)	SSIM	Inference time (ms)	Memory usage (MB)	Architectural contrast (%)	Energy (%)	Uniformity (%)
FA-SRGAN (Full)	32.79	0.926	143	1275	17.8	14.2	11.1
FA-SRGAN (w/o FLS)	31.58	0.91	137	1245	15.1	11.9	9.5
FA-SRGAN (w/o Attention)	31.91	0.914	139	1258	16.5	13	10.3
ESRGAN	31.27	0.902	158	1415	13.4	11.2	8.7
SwinIR	32.13	0.918	208	1842	15.7	12.6	9.2
Diffusion-based	32.41	0.922	477	2550	16.9	13.3	10

Table 7. Subjective evaluation by designers.

Method	Detail clarity	Visual comfort	Texture naturalness	Overall quality
FA-SRGAN (Full)	9.4	9.2	9.3	9.5
FA-SRGAN (w/o FLS)	8.6	8.5	8.7	8.8
FA-SRGAN (w/o Attention)	8.8	8.7	8.9	9
ESRGAN	8.2	8	8.1	8.3
SwinIR	8.9	8.6	8.8	9
Diffusion-based	9.1	8.9	9	9.2

contrast, energy, and uniformity improvements of 15%, 12%, and 10%, respectively, while SRGAN and VDSR have improvements of 10%, 8%, 6%, and 6%, 5%, and 4%, respectively. The data results indicate that FA-SRGAN can effectively enhance image details and significantly improve overall quality when processing high texture 3D animations.

4.6 Ablation experiment

A further comparison of the performance of different methods was conducted, along with ablation tests. The results are shown in Table 6.

Table 6 presents a comparative analysis of the proposed FA-SRGAN model and its variants against other state-of-the-art image super-resolution methods across multiple evaluation metrics. As shown in the data, the full FA-SRGAN model outperforms all compared methods in terms of both PSNR and SSIM, achieving an average PSNR of 32.79 dB and SSIM of 0.926, which are significantly higher than those of ESRGAN (31.27 dB / 0.902), SwinIR (32.13 dB / 0.918), and the diffusion-based method (32.41 dB / 0.922), demonstrating its superior performance in image reconstruction quality. Specifically, in architectural design scenes, FA-SRGAN improves contrast by 17.8%, energy by 14.2%, and uniformity by 11.1%, indicating that the method is particularly effective for enhancing high-texture regions in 3D animation design images and improving visual detail representation.

To further validate the effectiveness of each module within the model, two ablation experiments were conducted: one removing the fuzzy logic system (FA-SRGAN

w/o FLS) and another eliminating the attention mechanism (FA-SRGAN w/o Attention). The results show that without FLS, the PSNR drops to 31.58 dB and SSIM decreases to 0.91; similarly, removing the attention module leads to a PSNR of 31.91 dB and SSIM of 0.914, highlighting the positive impact of both components on image quality. Moreover, the inference time and memory usage reflect the practical deployment advantages of FA-SRGAN, with an average inference time of 143 ms and memory consumption of 1275 MB—substantially lower than SwinIR (208 ms / 1842 MB) and the diffusion-based method (477 ms / 2550 MB). This demonstrates that FA-SRGAN offers better real-time performance and resource efficiency, making it well-suited for efficient detail enhancement in complex 3D animation design workflows.

To further evaluate the actual visual performance of the proposed FA-SRGAN method in enhancing details in 3D animation design images, an experiment was conducted involving 15 professional 3D animators and visual designers. The evaluation criteria included detail clarity, visual comfort, texture naturalness, and overall quality, with a maximum score of 10 points for each category.

The results show that the full FA-SRGAN model achieves the highest scores across all subjective metrics, particularly in detail clarity and overall quality, indicating its superior ability to enhance visual details while maintaining natural appearance and user experience. The ablated versions [CE1] (without FLS or attention mechanism) still perform reasonably well but show noticeable degradation in subjective ratings, further confirming the importance of each component in contributing to the final visual outcome.

5 Conclusions

This article applied a 3D animation design image detail enhancement method by combining intelligent fuzzy algorithm and SRGAN. By utilizing the SRGAN model to process low resolution images, using a generator network to extract features and upsample, and applying a fuzzy logic system to dynamically adjust the image fuzzy degree, the ringing effect was successfully suppressed and the clarity and realism of the image were significantly improved. In addition, the integrated channel and spatial attention mechanism effectively focused on important areas, enhancing the performance of key details. The experimental results showed that Fuzzy Algorithm-SRGAN achieved an average PSNR of over 32.8 dB in various testing scenes, and improved contrast, energy, and uniformity by 18%, 14%, and 11%, respectively, in architectural design.

Meanwhile, the current experiments have primarily focused on synthetic or virtual scenes, without incorporating real-world industry applications. Future research can be further extended to practical industrial collaborations, such as conducting case studies in partnership with game development or film animation production teams, and applying the proposed method to real production workflows including character modeling, scene rendering, or visual effects processing. This would help validate the applicability of FA-SRGAN in handling complex dynamic scenes and high-precision texture requirements, while more clearly demonstrating its practical value in enhancing visual quality, optimizing rendering efficiency, and reducing manual post-processing costs. Through deep integration with the industry, this technology has the potential to transition from academic research toward real-world deployment, providing more intelligent and automated solutions for high-quality 3D content creation.

Acknowledgments

Not applicable.

Funding

This work was supported by Jiangsu Province Industry-University-Research cooperation project, "Design and development of panoramic display system of Zhou Enlai Memorial Hall based on VR technology", BY20230998.

Conflicts of interest

The authors declare that there are no conflicts of interest regarding the publication of this article.

Data availability statement

The data that support the findings of this study are available from the corresponding author upon reasonable request.

Author contribution statement

All authors listed have made a substantial, direct, and intellectual contribution to the work and approved it for publication.

References

1. A. Bakar Maharani, A. Juliansyah, R. Thalib, D.A. Pratama, High performances of stabilized Lanczos-types for solving high dimension problems: a survey[J], *Comput. Sci.* **16**, 837–854 (2021)
2. J. Panda, S. Meher, An improved image interpolation technique using OLA e-spline[J], *Egypt. Inform. J.* **23**, 159–172 (2022)
3. M. Ameen Musa, A.M. Ashir, M.S. Anwar, Diffraction-based ringing suppression in image deblurring[J], *Solid State Technol.* **64**, 3305–3313 (2021)
4. A. Acharya, S. Meher, Iterative spatial domain 2-D signal decomposition for effectual image up-scaling[J], *Multimed. Tools Appl.* **80**, 5577–5616 (2021)
5. R. Lan, L. Sun, Z. Liu, H. Lu, Z. Su, C. Pang et al., Cascading and enhanced residual networks for accurate single-image super-resolution[J], *IEEE Trans. Cybern.* **51**, 115–125 (2020)
6. M. Pharr, B. Wronski, M. Salvi, M. Fajardo, Filtering after shading with stochastic texture filtering[J], *Proc. ACM Comput. Graphics Interact. Techniq.* **7**, 1–20 (2024)
7. L. Bai, Y. Li, Z. Zhou, Visualization pipeline of autonomous driving scenes based on FCCR-3D reconstruction[J], *J. Electron. Imag.* **31**, 033023–033023 (2022)
8. M. Haris, G. Shakhnarovich, N. Ukita, Deep back-projection networks for single image super-resolution[J], *IEEE Trans. Pattern Anal. Mach. Intell.* **43**, 4323–4337 (2020)
9. Y. Chen, R. Xia, K. Yang, K. Zou, MFFN: image super-resolution via multi-level features fusion network[J], *Vis. Comput.* **40**, 489–504 (2024)
10. Y. Fu, Q. Yan, J. Liao, H. Zhou, J. Tang, C. Xiao, Seamless texture optimization for RGB-D reconstruction[J], *IEEE Trans. Vis. Comput. Graph.* **29**, 1845–1859 (2021)
11. K. Aarizou, A. Loukil, Self-similarity single image super-resolution based on blur kernel estimation for texture reconstruction[J], *Int. J. Comput. Sci. Eng.* **25**, 64–73 (2022)
12. W. Liu, P. Zhang, Y. Lei, X. Huang, J. Yang, M. Ng, A generalized framework for edge-preserving and structure-preserving image smoothing[J], *IEEE Trans. Pattern Anal. Mach. Intell.* **44**, 6631–6648 (2021)
13. R. Soundrapandian, S.C. Satapathy, C.M. Pvssr, N.G. Nhu, A comprehensive survey on image enhancement techniques with special emphasis on infrared images[J], *Multimed. Tools Appl.* **81**, 9045–9077 (2022)
14. Y. Wang, X. Li, F. Nan, F. Liu, H. Li, H. Wang et al., Image super-resolution reconstruction based on generative adversarial network model with feedback and attention mechanisms[J], *Multimed. Tools Appl.* **81**, 6633–6652 (2022)
15. A.-C. Tsai, C.-H. Tsou, J.-F. Wang, EESRGAN: Efficient & effective super-resolution generative adversarial network[J], *IETE Tech. Rev.* **41**, 200–211 (2024)
16. W. Zhang, Y. Wang, C. Li, Underwater image enhancement by attenuated color channel correction and detail preserved contrast enhancement[J], *IEEE J. Ocean. Eng.* **47**, 718–735 (2022)

17. S.W. Zamir, A. Arora, S. Khan, M. Hayat, F.S. Khan, M.-H. Yang et al., Learning enriched features for fast image restoration and enhancement[J], *IEEE Trans. Pattern Anal. Mach. Intell.* **45**, 1934–1948 (2022)
18. S. Yang, X. Wang, Sparse representation and SRCNN based spatio-temporal information fusion method of multi-sensor remote sensing data[J], *J. Netw. Intell.* **6**, 40–53 (2021)
19. J. Khandelwal, V.K. Sharma, W-VDSR: Wavelet-based secure image transmission using machine learning VDSR neural network[J], *Multimed. Tools Appl.* **82**, 42147–42172 (2023)
20. J. Daihong, Z. Sai, D. Lei, D. Yueming, Multi-scale generative adversarial network for image super-resolution [J], *Soft. Comput.* **26**, 3631–3641 (2022)
21. P. Wang, H. Zhu, H. Huang, H. Zhang, N. Wang, TMS-GAN: A twofold multi-scale generative adversarial network for single image dehazing[J], *IEEE Trans. Circ. Syst. Video Technol.* **32**, 2760–2772 (2021)
22. R. Abbas, N. Gu, Improving deep learning-based image super-resolution with residual learning and perceptual loss using SRGAN model[J], *Soft. Comput.* **27**, 16041–16057 (2023)
23. Z. Tang, T. Zhang, Y. Du, J. Su, Individual identification method of little sample radiation source based on SGDCGAN + DCNN[J], *IET Commun.* **17**, 253–264 (2023)
24. B. Rasti, Y. Chang, E. Dalsasso, L. Denis, P. Ghamisi, Image restoration for remote sensing: Overview and toolbox[J], *IEEE Geosci. Remote Sens. Mag.* **10**, 201–230 (2021)
25. D.Nh. Thanh, Vb.S. Prasath, L.M. Hieu, S. Dvoenko, An adaptive method for image restoration based on high-order total variation and inverse gradient[J], *Signal Image Video Process.* **14**, 1189–1197 (2020)
26. F. Fang, J. Li, Y. Yuan, T. Zeng, G. Zhang, Multilevel edge features guided network for image denoising[J], *IEEE Trans. Neural Netw. Learn. Syst.* **32**, 3956–3970 (2020)
27. X. Chen, S. Zhan, D. Ji, L. Xu, C. Wu, X. Li, Image denoising via deep network based on edge enhancement[J], *J. Ambient Intell. Humaniz. Comput.* **14**, 14795–14805 (2023)
28. S. Izadi, D. Sutton, G. Hamarneh, Image denoising in the deep learning era[J], *Artif. Intell. Rev.* **56**, 5929–5974 (2023)
29. R.S. Jebur, M.H.B.M. Zabil, D.A. Hammood, L.K. Cheng, A comprehensive review of image denoising in deep learning[J], *Multimed. Tools Appl.* **83**, 58181–58199 (2024)
30. M. Kumar, S.K. Mishra, A comprehensive review on nature inspired neural network based adaptive filter for eliminating noise in medical images[J], *Curr. Med. Imag.* **16**, 278–287 (2020)
31. Y. Qiu, Y. Yang, Z. Lin, P. Chen, Y. Luo, W. Huang, Improved denoising autoencoder for maritime image denoising and semantic segmentation of USV[J], *China Commun.* **17**, 46–57 (2020)
32. S. Yadav, S.K. Saha, R. Kar, D. Mandal, Noise confiscation from sEMG through enhanced adaptive filtering based on evolutionary computing[J], *Circuits Syst. Signal Process.* **42**, 4096–4128 (2023)
33. S. Zhang, H. Liu, M. Hu, A. Jiang, L. Zhang, F. Xu et al., An adaptive CEEMDAN thresholding denoising method optimized by nonlocal means algorithm[J], *IEEE Trans. Instrum. Meas.* **69**, 6891–6903 (2020)
34. Z. Bao, G. Zhang, B. Xiong, S. Gai, New image denoising algorithm using monogenic wavelet transform and improved deep convolutional neural network[J], *Multimed. Tools Appl.* **79**, 7401–7412 (2020)
35. J.A. Iglesias-Guitian, P. Mane, B. Moon, Real-time denoising of volumetric path tracing for direct volume rendering[J], *IEEE Trans. Vis. Comput. Graph.* **28**, 2734–2747 (2020)
36. J. Liu, B. Hui, K. Li, Y. Liu, Y.-K. Lai, Y. Zhang et al., Geometry-guided dense perspective network for speech-driven facial animation[J], *IEEE Trans. Vis. Comput. Graph.* **28**, 4873–4886 (2021)

Cite this article as: Haitao Pu, Yuang Pu, 3D animation design image detail enhancement based on intelligent fuzzy algorithm , *Int. J. Simul. Multidisci. Des. Optim.* **16**, 9 (2025), <https://doi.org/10.1051/smdo/2025010>